**< > = exemplo**

**Acessar**

* **Acessar o MYSQL= mysql -h localhost -u root**

**Criar**

* **Criar tabela = create table <tabela>(nome tabela e tipo de dado);**
* **Criar banco de dados= create database;**
* **Criar tabela já com referência de chave estrangeira = create TABLE <tabela> (<coluna e tipo de dado> primary KEY AUTO\_INCREMENT, CONSTRAINT FOREIGN KEY(<coluna>) REFERENCES <tabaela>(<coluna>));**

**Inserir**

* **Inserir chave primária após criar coluna = ALTER TABLE <turma> ADD PRIMARY KEY(<coluna>);**
* **inserir chave estrangeira = alter table <tabela receberá chave estrangeira> add constraint foreign key (<coluna receberá chave estrangeira>) references < tabela que faz referência> (<coluna que faz referência>);**
* **Inserir uma coluna na tabela = alter table <nome da tabela> add <nova coluna e tipo de dados>;**
* **Inserir um atributo especificando antes e depois = alter table <tabela> add <qual coluna a ser adicionado e tipo de dado> after <coluna>;**
* **Inserir informações na tabela = insert into <tabela> values(null <para deixar vazio> ,"jose da Silva","111.222.333-44", "2023-03-09");**
* **inserir o campo auto incrementado - alter table <tabela> modify <nome coluna e dados> AUTO\_INCREMENT;**
* **Inserir informação entre dados - insert into <tabela> (<colunas>) values("111.222.333-44",2);**
* **INSERT INTO <tabela> VALUES (1, "Carlos"), (2, "Maria"), (3, "José"), (4, "Julia"), (5, "Chico")**

**Visualizar**

* **Mostrar histórico de atualização da tabela = show create table;**
* **Mostrar tabela= show tables <nome da tabela>;**
* **Quando dentro do banco, mostrar tabelas = show tables;**
* **Visualizar detalhes da relação entre chaves estrangeiras = show create table <tabela de chaves estrangeira>;**
* **Mostrar informações alocadas na tabela= select \* from <tabela>;**
* **Exibir detalhes dos valores de uma tabela - SELECT \* from <tabela>**
* **Mostrar em tabela a ligação da chave primária com chave estrangeira = select \* from cliente join cliente\_compra\_produto;**

**select cliente.nome, cliente.cpf, cliente\_compra\_produto.cod\_produto from cliente join cliente\_compra\_produto on cliente.cpf = cliente\_compra\_produto.cliente\_cpf;**

**Selecionar**

* **Selecionar o banco de dados= use <nome do banco>;**
* **Selecionar uma coluna específica = select <nome da coluna, para vários separa-se a coluna por virgula> from cliente;**

**Alterar**

* **Alterar o tipo de dados da coluna= alter table <tabela> modify <coluna e tipo de dados>;**
* **Mudar nome da tabela= alter table <tabela> change <nome\_atual nome\_novo tipo de dado>;**

**Deletar**

* **Deletar uma <coluna>= alter table <tabela> drop <coluna>;**
* **Deletar tabela = delete from <tabela> where <coluna> = <linha de identificação>;**
* **Deletar chave estrangeira = ALTER TABLE <tabela> DROP FOREIGN KEY <curso\_ibfk\_1>**
* **Deletar valor em tabela = DELETE from <tabela> WHERE <coluna> = <posição>**
* **Deletar valor vinculado a chave = delete from carro where cod\_cliente = (select cod\_cliente from cliente where nome ="Carla Pinho");**
* **Deletar tabela View = drop view <tabela>;**

**Backup**

* **Fazer backup do banco de dados = mysqldump -h localhost -u root loja > <nome arquivo.sql**

**Busca**

* **Buscar por informação específica = select \* from <tabela> where cpf="111.222.333-44";**
* **Busca por maior que e menor que = select \* from <tabela> where <coluna> = 50;**
* **Buscar palavra com partes = select \* from <tabela> where <coluna> like "%fei%"; <like= como,semelhante / % = indefinido>**
* **busca com operadores condicional = select <coluna> from <tabela> where estoque >= 70 and valor <=100;**
* **Busca de dados entre dados= select \* from <tabela> where <coluna> between 1 and 4;**

**Operadores**

* **Realiza contagem de registro da coluna = select count(<nome>) from <tabela>;**
* **Visualizar contagem renomeando a coluna momentaneamente = select count (nome) as <nome desejado> from cliente;**
* **Somar valores e exibir em uma tabela nomeada = select sum(<valor>) as <nome da tabela> from produto;**
* **Exibir cálculo de média = select avg(<valor>) as <renomear tabela temporariamente> from <tabela>;**

**Abrir**

* **Abrir arquivo no MySQL = mysql -h localhost -u root < <nome do arquivo>.sql**

**Ordenar**

* **Ordenar a coluna específica = select \* from <tabela> order by <coluna> desc ou asc;<decrescente e crescente>**

**Habilitar**

* **Habilitar a gravação de registros no backup = tee <nome do arquivo>.txt**
* **Desativar a gravação do registro = notee**

**Atualizar**

* **Atualizar tabela- update <tabela> set valor = 4.85 where <coluna>=4;**
* **Modificando de várias colunas = update carro set cor = "roxo", modelo = "kwid" where placa = "ddd4d44";**

**Chave estrangeira**

**Criar evento de cascata para deletar dados que fazem referências a muitas tabelas :**

* **Cascate = ALTER TABLE <tabela> ADD CONSTRAINT FOREIGN KEY (<coluna>) REFERENCES <tabela>(<coluna>) on UPDATE CASCADE / ON DELETE CASCADE**
* **SET NULL= alter TABLE <tabela> add CONSTRAINT FOREIGN KEY(<coluna>) REFERENCES <tabela>(<coluna>) on UPDATE SET null on DELETE set null**
* **- Mostrar em tabela a ligação da chave primária com chave estrangeira = select \* from cliente join cliente\_compra\_produto;**

**- select cliente.nome, cliente.cpf, cliente\_compra\_produto produto.cod\_produto from cliente join cliente\_compra\_produto on cliente.cpf = cliente\_compra\_produto.cliente\_cpf;**

**OBS:**

* **Sprint = desenvolvimento de uma função pela equipe.**
* **SQL Lite = site para uso do SQL.**

**View (tabela virtual)**

**create view cliente\_view as select cliente.nome as nome\_cliente, produto.nome from cliente join cliente\_compra\_produto on cliente.cpf = cliente\_compra\_produto.cliente\_cpf join produto on produto.cod\_produto = cliente\_compra\_produto;**

**mostrar informações com tabelas relacionadas = select carro.placa,cliente.nome from cliente join carro on cliente.cod\_cliente = carro.cod\_cliente;**

**Função**

**Criar delimitador = delimiter <simbolo para delimitar>**

**Criar procedimento = create procedure <nome da função>(<parâmetro com valor de dados>)**

**- begin**

**- comandos**

**- end**

**EX:**

**IN = inserir dados**

**OUT = Não inserir informação**

**- Inserir valor pela função = call add\_produto(7, "cafe", 46, 6.54);**

**- Deletar o procedimento/função = drop procedure add\_produto;**

**- atualizar dados na função = create procedure atualizar(in cod int, in estoque int)**

**-> begin**

**-> update produto set <coluna> = estoque where cod\_produto = cod;**

**-> select \* from produto;**

**-> select sum(estoque) as total from produto;**

**-> end;;**

**ACL (Acces Control level)**

**criar usuário em servidor = create user "gerente"@"localhost" identified by "123";**

**Dar acesso total usuário = grant all privileges on loja.\* to "gerente"@"localhost";**

**Dar acesso limitado = grant <comandos permitidos> on loja.produto to "estagiario"@"localhost";**

**Dar acesso a todos os comando de uma tabela = grant all privileges on loja.cliente to "engenheiro"@"localhost";**

**informações de acesso = show grants for "gerente"@"localhost";**

**remover acesso específico = revoke all privileges on loja.\* from "gerente"@"localhost";**

**remover usuário = drop user "engenheiro"@"localhost";**

**mudar senha usuário = SET PASSWORD FOR 'usuário'@'local' = PASSWORD('nova senha');**